Sentiment Analysis using Twitter Data

1st Gautam Ahuja
Computer Science
Stevens Institute of Technology
Jersey City, USA
gahuja@stevens.edu

2nd Shoaib Kalawant
Computer Science
Stevens Institute of Technology
Jersey City, USA
skalawan@stevens.edu

3rd Sanjana Gupta
Data Science
Stevens Institute of Technology
Jersey City, USA
sgupta58@stevens.edu

I. Introduction

Briefly introducing the problem of entity-level sentiment analysis on Twitter data is crucial for understanding the context and significance of the project. The pervasive use of social media platforms like Twitter has led to an abundance of unstructured textual data, making sentiment analysis a vital task. This project focuses on analyzing sentiments expressed in tweets concerning specific entities. The primary objectives include discerning positive, negative, and neutral sentiments associated with entities mentioned in tweets.

A. Significance of the Problem

Highlighting the importance of entity-level sentiment analysis, particularly on Twitter, is essential. The sentiment expressed towards entities on social media can influence public opinion, impact brand perception, and provide valuable insights into user behavior. Understanding and categorizing sentiments associated with entities enable businesses, researchers, and policymakers to make informed decisions based on public reactions.

B. Methodology Overview

Provide a concise summary of the methodology adopted for entity-level sentiment analysis. This includes data collection, preprocessing, and the application of machine learning algorithms. The choice of the Multinomial Naive Bayes Classifier is emphasized due to its suitability for processing textual data and its initial implementation success.

II. Related Work

A. Naive Bayesian Model

Assume you wish to categorize Twitter reviews as positive, negative, or neutral. Sentiment analysis is a common task performed by data scientists to understand the sentiment expressed in textual data. The Naive Bayes algorithm is a straightforward and fast classification algorithm widely employed for large datasets. It has found success in various applications such as spam filtering, text classification, sentiment analysis, and recommendation systems. The Naive Bayes classifier utilizes Bayes' probability theorem for predicting unknown class labels.

For our Twitter sentiment analysis, we'll categorize the reviews into three classes: positive, negative, and neutral. The algorithm calculates the probability of a review belonging to a specific class based on the observed frequencies of words.When applied to textual data like Twitter reviews in Natural Language Processing (NLP), Naive Bayes classification can face reliability challenges with larger datasets. The algorithm's assumption of feature independence may limit its effectiveness. While still popular and efficient for moderate-sized datasets, consideration of alternative, more sophisticated models is crucial for improved accuracy with extensive and diverse textual data.

B. SENTIMENT ANALYSIS WITH LONG SHORT-TERM MEMORY NETWORKS

Recurrent Neural Networks (RNNs) and Long Short-Term Memory networks (LSTMs) enhance sentiment analysis on Twitter. By introducing memory elements, these models capture short-term dependencies, crucial for understanding sentiment in the rapidly changing context of tweets.

In the context of sentiment analysis on Twitter data, a Long Short-Term Memory (LSTM) model has proven effective in addressing the limitations of traditional Recurrent Neural Networks (RNNs). The LSTM's ability to handle long-term dependencies is crucial for accurately gauging sentiment over the course of a tweet. Twitter language often involves complex structures with extended contextual dependencies, and the LSTM's long-term memory feature significantly improves the model's performance. This ensures a more comprehensive analysis of sentiment, enabling the model to consider not only immediate context but also capture the nuanced sentiment patterns that may unfold over the entire length of a tweet.

However, challenges persist. Long-term dependencies in Twitter language can pose hurdles, impacting the models' ability to fully grasp intricate sentiment nuances. Both RNNs and LSTMs primarily rely on preceding context, potentially limiting their effectiveness in capturing the complete sentiment dynamics in the diverse landscape of Twitter data. Alternative approaches may be explored to enhance the models' adaptability to both short-term and long-term dependencies.

III. SOLUTION

We have initiated our sentiment analysis project by implementing a Naive Bayes classifier for Twitter data. Prior to the implementation phase, extensive preprocessing of the dataset was conducted. This preprocessing phase involved tasks such as tokenization, stop-word removal, and handling emojis and hashtags to ensure the data is well-structured and ready for analysis. By incorporating these preprocessing steps, we aim to enhance the effectiveness of our Naive Bayes classifier in capturing meaningful patterns and sentiments from the Twitter dataset.

A. Description of the dataset

The training dataset utilized in this project is obtained from Twitter and encompasses four key attributes: ’id,’ ’land,’ ’sentiment,’ and ’tweet.’ Beyond these fundamental features, we have introduced a label category that assigns distinct numerical values to sentiment classes, mapping ’positive’ to 1, ’negative’ to 2, ’neutral’ to 3, and ’irrelevant’ to 3. This curated dataset is meticulously designed to facilitate entity-level sentiment analysis on Twitter data, providing a rich set of information including message content, associated entities, sentiment labels, and additional attributes. The incorporation of the label category enhances the dataset’s versatility, allowing for a more nuanced exploration of sentiment patterns and enabling a comprehensive analysis of tweets in a variety of contexts.

B. Machine Learning Algorithms

- Naive Bayesian Model:
  Naive Bayes is one of the most straightforward and fast classification algorithms. It is very well suited for large volumes of data. It is successfully used in various applications such as : Spam filtering, Text classification, Sentiment analysis, Recommender systems.

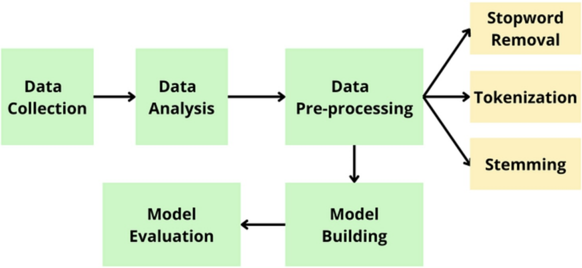

Fig. 1. Model Steps

  The expression below is the fundamental formula for conditional probability using the Naive Bayes classifier. It represents the probability of a particular label given a set of features (f1, f2, f3, ..., fn). It uses the Bayes theorem of probability for the prediction of unknown classes.

$$p(\text{label}|f_1, f_2, \ldots, f_n) = \frac{p(f_1|\text{label}) \cdot p(f_2|\text{label}) \cdot \ldots \cdot p(f_n|\text{label}) \cdot p(\text{label})}{p(f_1, f_2, f_3, \ldots, f_n)}$$

  This formula assumes the Naive Bayes assumption that the features are conditionally independent given the label, simplifying the calculation. The goal is often to find the label that maximizes this conditional probability, making Naive Bayes an efficient and widely used classification algorithm.
- Bi-LSTM
  The Long Short-Term Memory (LSTM) network is effective in capturing sequential information for sentiment analysis but struggles with polysemous words in different contexts. To address this, we propose a Bidirectional LSTM (BiLSTM) model. This BiLSTM model incorporates topic information, enabling a more nuanced representation of polysemous words within specific contexts. By learning topic-based representations, the model automatically captures the meaning of polysemous words and long sequential information, enhancing its sentiment analysis capabilities.
  LSTMs, a specialized class of Recurrent Neural Networks (RNNs), excel at retaining information over extended periods. The introduction of Bidirectional LSTMs further enhances their utility in text classification tasks. Unlike unidirectional models, Bidirectional LSTMs consider contextual information in both forward and reverse directions, providing a more comprehensive understanding of the text. This bidirectional approach proves valuable in tasks like text classification, where capturing context from both ends is beneficial.
  In a simplistic explanation of Bidirectional RNN, each RNN cell functions as a black box, taking a hidden state and a word vector as input and producing an output vector along with the next hidden state. Weight-sharing occurs across all words in the sentence, and these weights are tuned through Backpropagation of the losses. In the Bidirectional RNN, text is read in both the usual and reverse fashion by stacking two RNNs in parallel. This results in eight output vectors that capture information from both directions, facilitating a more thorough representation of the text for sentiment analysis.

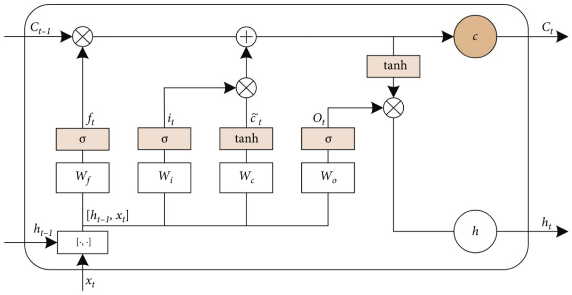

Fig. 2. Model Steps

C. Solution

Before starting implementing we need to preprocess the data and depending on how well the data has been preprocessed; the results are better.

In NLP, text preprocessing is the first step in the process of building a model. The various text preprocessing steps used in our model are:

- Tokenization: Tokenization is a step which splits longer strings of text into smaller pieces, or tokens. Larger chunks of text can be tokenized into sentences, sentences can be tokenized into words, etc.
- Clean numbers: Since we are dealing with text, so the number might not add much information to text processing. So, numbers can be removed from text.
- Clean Special Characters: Special characters, as you know, are non- alphanumeric characters. These characters are most often found in tweets, references, currency numbers etc.
- Remove Stopwords: we efficiently removed irrelevant stopwords to enhance the quality of the text data. This step focuses on eliminating non-informative words, streamlining the dataset for improved sentiment analysis accuracy.

Implementaion of Naive Bayes Algorithm:

In the implementation of the Naive Bayes algorithm, we applied the multinomial variant post the preprocessing steps detailed earlier. The training phase involved utilizing the provided training set to educate the model on the intricacies of sentiment patterns within the Twitter data. After training, the model demonstrated commendable performance on the training set, achieving an accuracy of approximately 78.23%. The corresponding F1 score, a metric that balances precision and recall, also reflected a robust performance, hovering around 78%. These results affirm the model's ability to effectively capture sentiment nuances and generalize well on the training data.

Moving to the testing phase, the trained Naive Bayes model was evaluated on a separate testing set to assess its performance on previously unseen data. The testing accuracy was noted to be around 61%, indicating the model's ability to generalize to new instances. Moreover, the F1 score on the test data stood at approximately 67%, affirming the model's competence in handling sentiment classification tasks beyond the training set. These results collectively demonstrate the Naive Bayes algorithm's proficiency in sentiment analysis on Twitter data, offering valuable insights into its generalization capabilities and performance on unseen instances.

Next Steps:

As we embark on the implementation of Recurrent Neural Networks (RNN) for our sentiment analysis project, our future trajectory is shaped by the need to overcome the observed challenges, particularly the lower accuracy associated with the Multinomial Naive Bayes model. The incorporation of RNNs, renowned for their proficiency in capturing sequential dependencies, holds promise in elevating the accuracy of sentiment analysis on Twitter data. This shift towards RNNs aims to enhance the model's ability to discern the contextual intricacies embedded in tweets, surpassing the limitations posed by the simplistic Naive Bayes approach. Our strategic plan involves meticulous fine-tuning of the RNN model, exploring diverse architectures, and leveraging the inherent sequential structure of language to augment accuracy in sentiment classification. Furthermore, our future efforts will be dedicated to conducting thorough evaluations, incorporating metrics such as precision, recall, and F1 score, to holistically gauge the RNN model's efficacy in comparison to the Naive Bayes methodology. This strategic pivot underscores our commitment to continuous improvement, ensuring heightened accuracy and resilience in capturing nuanced sentiment patterns within the Twitter data landscape.